

Many clients do not have one to one or one to many assignment of roles to role owners rather they have some specific logic to define role owners. In our setup, GRC approvers are defined based on functional area, business process, connector etc. For easy maintenance of GRC approvers role-based agent rule was created.

Resolution: Create a BRF+ Agent rule that will fetch approvers, based on decision table criteria and dummy role assignment.

#### BRF+ Design:

Below Steps will be followed to achieve required results:

Step 1: Create Structure "AGR\_USERS"

Step 2: Create a decision table to provide "dummy role name" based on suitable selection.

Step 3: Create a DB lookup on table "AGR\_USERS" to fetch users assigned to appropriate dummy roles.

Step 4: Create a "Ruleset" to process DB lookup and return approvers.

#### BRF+ Configuration:

You have to generate the BRF+ Rule via Transaction SPRO in GRC system. Follow the below steps in your GRC system.

Run the transaction SPRO, go to IMG => Governance, Risk and Compliance =>Access Control =>Workflow for Access Control => Define Workflow related MSMP rules.

Or

**Directly execute TCode GRFNMW\_DEV\_RULES**

Fill generation criteria (Process ID, Rule type, etc.)

Specify Generation options

Generate rule shell (Execute button)

After successful rule generation, goto BRF+ to check newly created BRFPlus Application

The screenshot shows the 'Rule Info' and 'Generation of Options' sections of a BRF+ configuration screen. The 'Rule Info' section contains the following fields:

MSMP Process ID	SAP_GRAC_ACCESS_REQUEST
Rule Kind	Agents Rule
Rule ID	ZBRF_ROLEOWNER
Application/Func. Group Name	ZBRF_ROLEOWNER
Rule Type	BRFplus Flat Rule (Linitem by Linitem)

The 'Generation of Options' section contains the following fields:

<input checked="" type="checkbox"/> Generate Rule	
<input checked="" type="checkbox"/> Gen. Result Work-area (BRF+)	
Override BRF+ Application Text	
Override BRF+ Function Text	

#### Function Signature update:

In BRF+ function, change the mode to "Event Mode" and activate the function as shown below.

Since Function mode has been changed to "Event mode," the result data object has changed automatically, so it has to be reset manually

In "Signature" tab of BRF Function, change the result data object to GRFN\_MW\_T\_AGENT\_ID

Simulation
Traces
Generate
Create Code Template
Advanced Checks

Mode:
Event Mode

Signature
Assigned Rulesets
Code Generation

### Context

Add Existing Data Object
Add New Data Object
Remove

Component Name	Text	Type
> GRAC_S_REQUEST_RULE_HEADER	Request Header	Structure
> GRAC_S_REQUEST_RULE_LINE	Access Request Line Item	Structure
> GRFN_MW_S_AGENT_ID	Result - Agent Rule	Structure

Result Data Object

Data Object:
GRFN\_MW\_T\_AGENT\_ID

### Create Structure:

From context menu of BRF+ application, create a Structure and bind it to AGR\_USERS.  
Save and activate this structure.

Create Structure

### General data

Type:	Structure
* Name:	AGR_USERS
Short Text:	AGR_USERS
Text:	AGR_USERS
Application:	ZBRF_ROLEOWNER

### Structure

Define Data Binding

Binding Type:	Bind to Structure Type (DDIC)
DDIC Type Name:	AGR_USERS

Where-Used List

Create
Create And Navigate To Object
Cancel

### Create a Decision Table:

From context menu of BRF+ application, create an Expression of type "Decision Table"

## Create Decision Table



Type:

\* Name:

Short Text:

Text:

Application:

Is Reusable: ☒

Create

Create And Navigate To Object

Cancel

Add "Condition" as well as "Result" column based on requirement

Note that AGR\_NAME in result column can only be selected if structure AGR\_USERS has already been created.

### List of Columns

#### Condition Columns

	Column Name	Text	Mandatory Input	Column Accessibility
<input type="radio"/>	FUNCAREA	FUNCAREA	<input type="checkbox"/>	Full Access (Changes Allowe
<input type="radio"/>	BPROC	BPROC	<input type="checkbox"/>	Full Access (Changes Allowe
<input type="radio"/>	DEPARTMENT	DEPARTMENT	<input type="checkbox"/>	Full Access (Changes Allowe
<input checked="" type="radio"/>	CONNECTOR	CONNECTOR	<input type="checkbox"/>	Full Access (Changes Allowe

#### Result Columns

	Column Name	Text	Action Column	Mandatory Input	Column Accessibility
<input checked="" type="radio"/>	AGR_NAME	AGR_NAME	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Full Access (Changes /

Create dummy roles in GRC backend system and assign to approvers

Populate the decision table with business data as required

Table Contents				
Find: <input type="text"/>		Next	Previous	Table Settings
<input type="checkbox"/>	FUNCAREA	BPROC	DEPARTMENT	CONNEC...
<input type="checkbox"/>	=31.01 (031.01...	=FSCM (FSCM)		[*031*]
<input type="checkbox"/>	=31.01 (031.01...	=LO-AB (LO-AB)		[*031*]
<input type="checkbox"/>	=31.01 (031.01...	=SSC (SSC/Shared Service Center)		[*031*]
<input type="checkbox"/>	=31.02 (031.02...	=FI (Finance+Controlling)		[*031*]
<input type="checkbox"/>	=31.02 (031.02...	=FSCM (FSCM)		[*031*]
				AGR_NAME
				CSOA-CDE-GRAC-RAP-031.01-FSC...
				CSOA-CDE-GRAC-RAP-031.01-LOA...
				CSOA-CDE-GRAC-RAP-031.01-SSC...
				CSOA-CDE-GRAC-RAP-031.02-FI (...)
				CSOA-CDE-GRAC-RAP-031.02-FSC...

Create DB Lookup: From context menu of BRF+ application, create an Expression of type "DBLookup"

Create Database Lookup

Type: Database Lookup

\* Name: GET\_APPROVER\_LOOKUP

Short Text: GET\_APPROVER\_LOOKUP

Text: GET\_APPROVER\_LOOKUP

Application: ZBRF\_ROLEOWNER

Is Reusable: ☒

Create

Create And Navigate To Object

Cancel

Add below details to the DB Lookup, here we are applying DB lookup based on result received from ZROLEOWNER on table AGR\_USERS and inserting user ID into GRFN\_MW\_T\_AGENT\_ID

**Save and generate the DB Lookup**

Create Database Lookup

Type: Database Lookup

\* Name: GET\_APPROVER\_LOOKUP

Short Text: GET\_APPROVER\_LOOKUP

Text: GET\_APPROVER\_LOOKUP

Application: ZBRF\_ROLEOWNER

Is Reusable: ☒

Create

Create And Navigate To Object

Cancel

Create Ruleset:

Click "Create and Navigate To Object"

**Create Rule as below:**

Create Ruleset

Type: Ruleset

\* Name: GET\_APPROVER

Short Text: GET\_APPROVER


Text: GET\_APPROVER

Application: ZBRF\_ROLEOWNER

Create

Create And Navigate To Object

Cancel


Create Rule 

---






**Disable Rule** Valid from  until


Description:

---


**If** Assign Condition... 

**Then**

(1) Change  GRFN\_MW\_T\_AGENT\_ID  after processing  GET\_APPROVER\_LOOKUP  Options 


**Else** Add 



**OK**




**Detail** 

Show Ruleset Header Context Overview

---



**Rules** 

Insert Rule  Insert Exit Condition 


 **Disable Rule** (1) Rule: No description is available - Rule has not been defined Options  

Valid From / To:  /

Description:



(1) Change  GRFN\_MW\_T\_AGENT\_ID after processing  GET\_APPROVER\_LOOKUP


Assign the ruleset to the function

**Detail** 

Hide Ruleset Header Context Overview

Enabled: ☒

Function:  ZBRF\_ROLEOWNER 

Precondition: <Not assigned> 


Number of Rules:

Number of Variables:


Priority:

---


**Variables**



(1) <Not assigned> 

**Expressions Which Initialize the Variables**

(1) <Not assigned> 

---

**Rules** 

Insert Rule  Insert Exit Condition 

Go to the function and under assigned ruleset you should be able to find "GET\_APPROVER" ruleset.

Detail

Simulation Traces Generate Create Code Template Advanced Checks

Mode: Event Mode

Signature Assigned Rulesets Code Generation

### Ruleset

Create Ruleset

Name	Text	Priority	Enabled	Precondition	Status	Executable
<input type="checkbox"/> GET_APPROVER	GET_APPROVER	[undefined]	Yes			✓

Simulate BRF+ rule:

Provide mandatory context values like Business Process, Functional Area, Connector.

Result

GRFN\_MW\_T\_AGENT\_ID

LINE_ITEM_KEY	USER_ID	NOTIFY_EXT_WHO_TYPE	NOTIFY_EXT_WHO_ID
**	SAKAUSAR	**	**

The function should return users IDs of the role owner as per the settings in decision table.

**Conclusion:** This custom agent rule will allow us to skip standard role owner maintenance process in GRC and simply assign approvers by assigning them dummy roles created in GRC system. Dummy roles can be created as per our requirement for example based on system\client, departments, business process etc.